Fig. 1

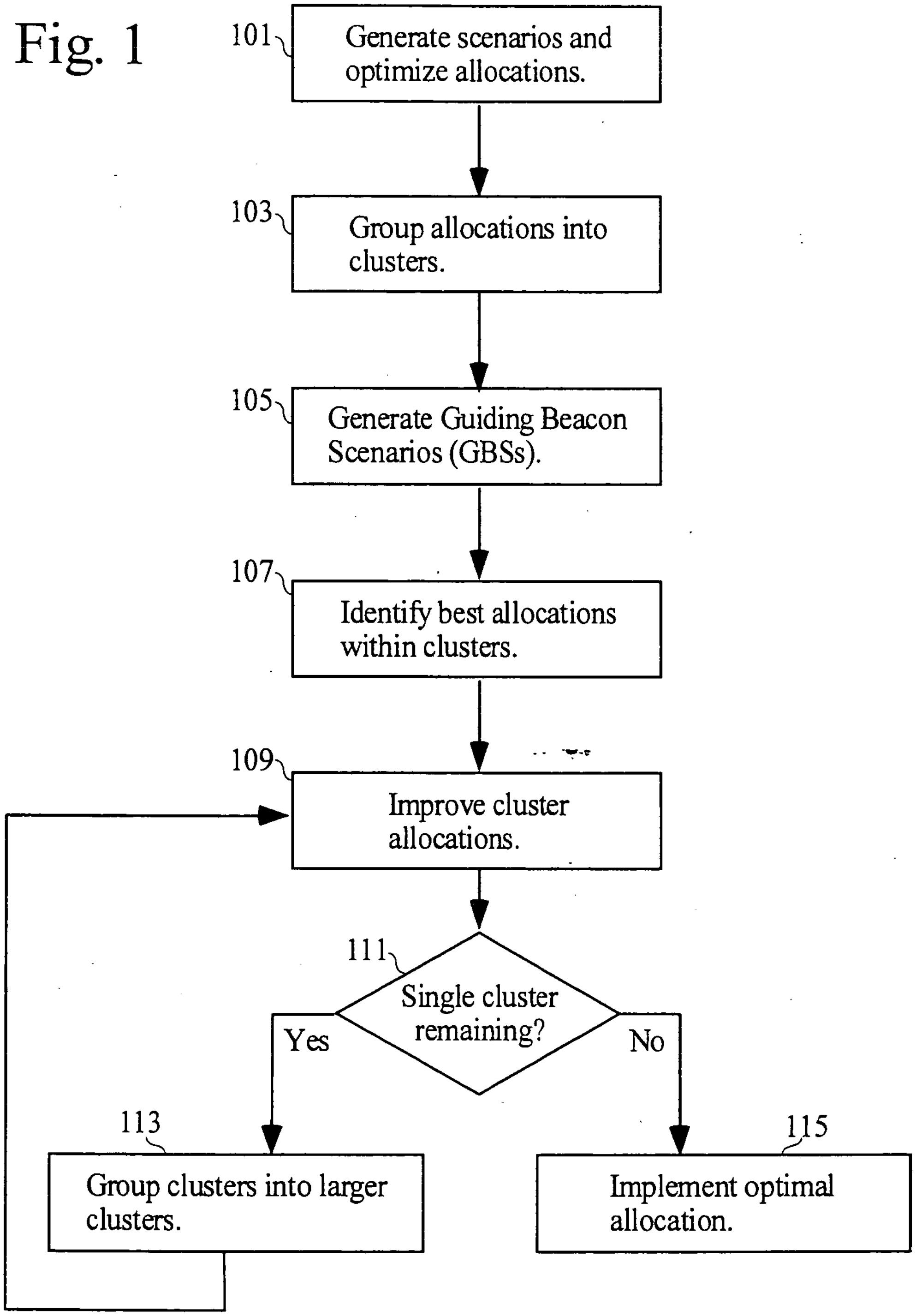

101
Generate scenarios and optimize allocations.
103
Group allocations into clusters.
105
Generate Guiding Beacon Scenarios (GBSs).
107
Identify best allocations within clusters.
109
Improve cluster allocations.
111
Single cluster remaining?
Yes
No
113
Group clusters into larger clusters.
115
Implement optimal allocation.

Fig. 2

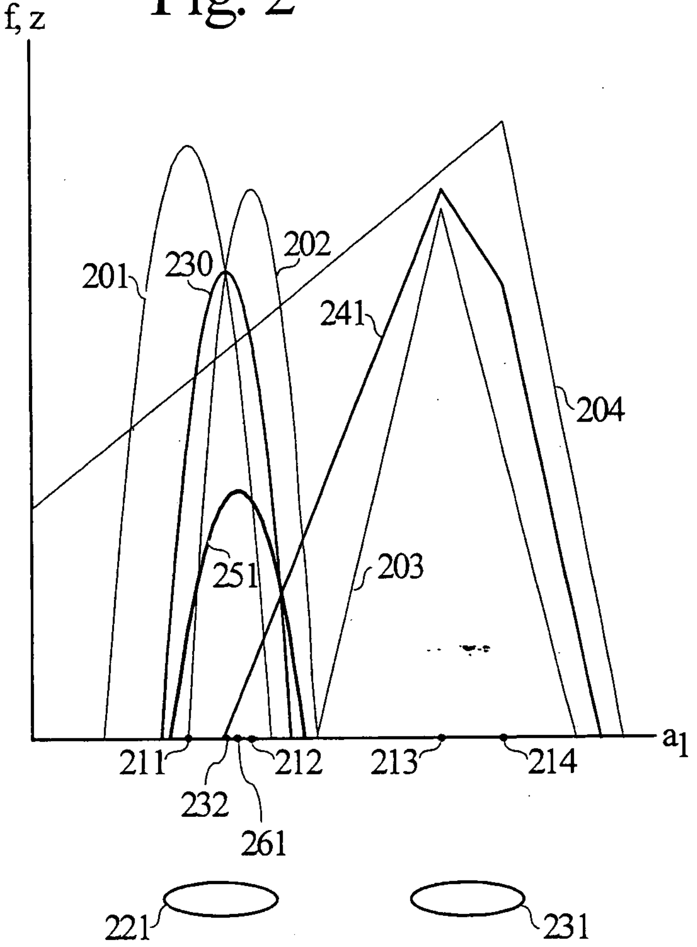


Fig. 3A

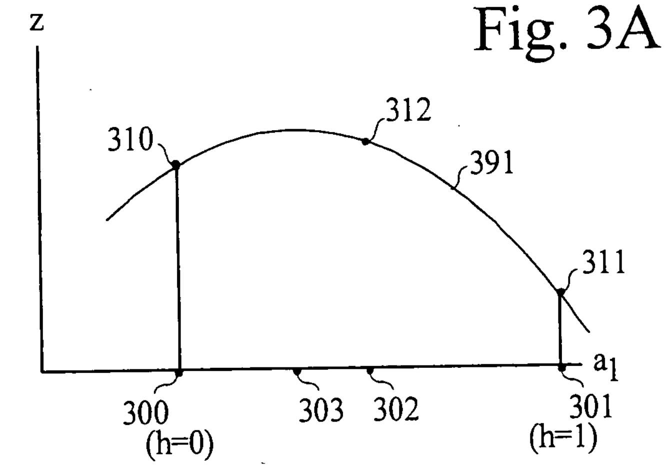


Fig. 3B

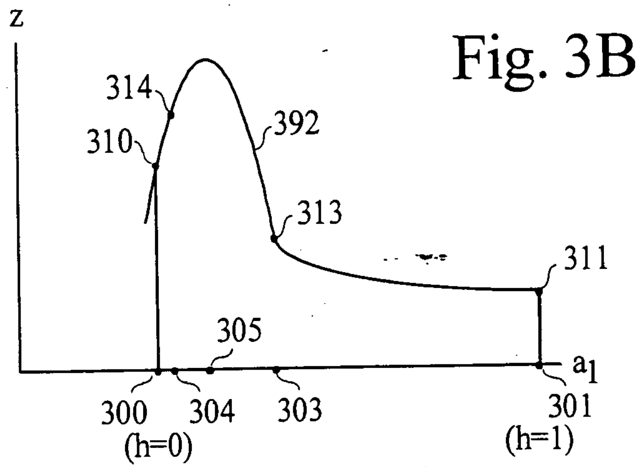


Fig. 3C

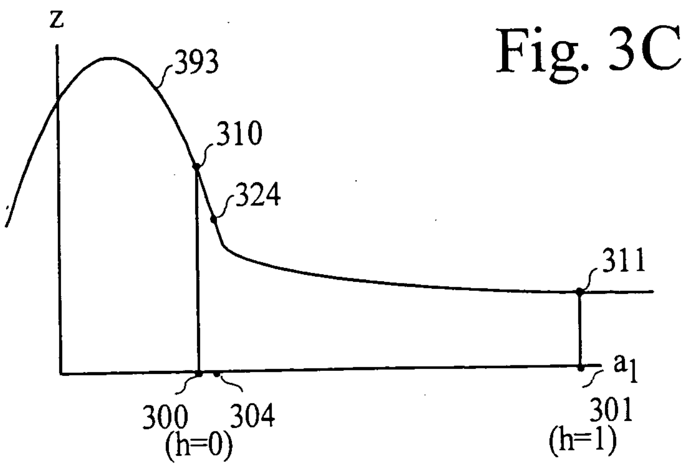


Fig. 4A

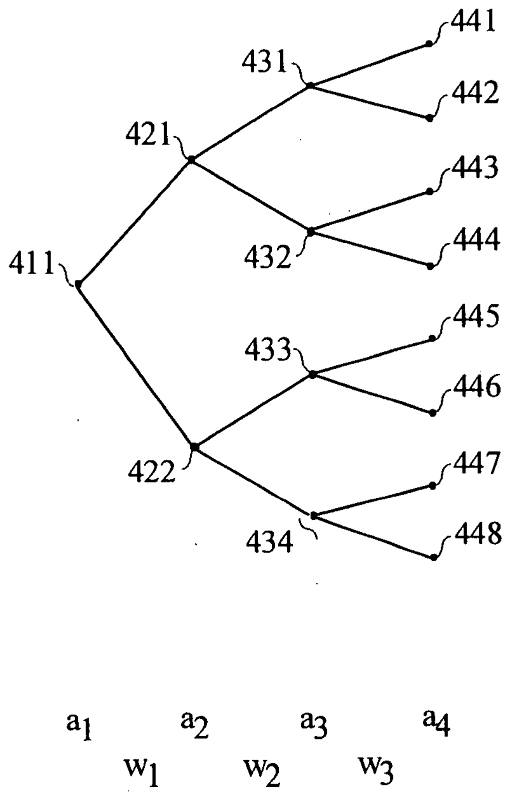

441
442
443
444
445
446
447
448
431
432
433
434
421
422
411
a_1
a_2
a_3
a_4
w_1
w_2
w_3

Fig. 4B

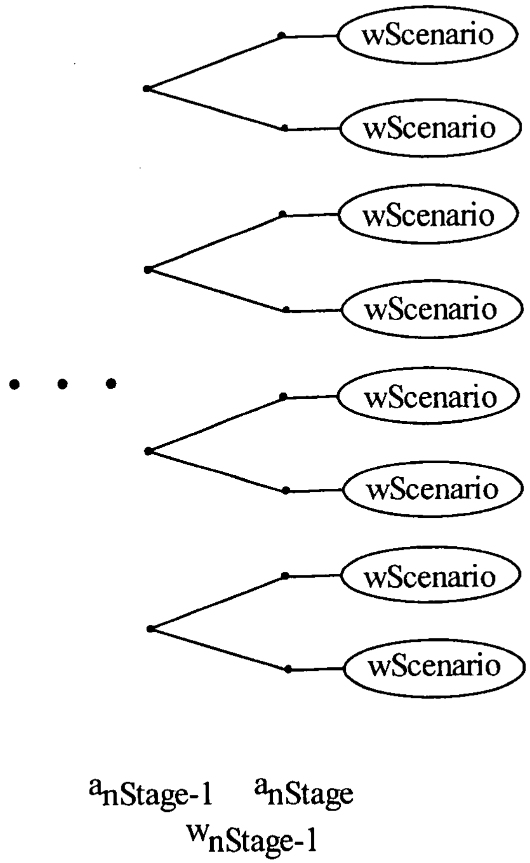

wScenario
wScenario
wScenario
wScenario
wScenario
wScenario
wScenario
wScenario
$a_{nStage-1}$
a_{nStage}
$w_{nStage-1}$

Fig. 5

```
class WNode : public...
    {
    pGBS;
    pWScenario;
    xAllocRndCntn;
    nodeProbability;
    voa;
    nextWNodeCntn;
    };
```

Fig. 6

```
class WScenario : public...
    {
    wwMatrix;
    pWNode[nStage+1];
    nativeProbability;
    nativeXAlloc;
    NativeOptimizer();
    };
```

Fig. 7

```
class XAlloc : public...
    {
    al[];
    feasible;
    h;
    voa;
    operator=( XAlloc& s);
    operator==(XAlloc& r);
    operator!=(XAlloc& r);
    };
```

Fig. 8A

```
class WWMatrix : public...
    {
    ww[][nStage];
    h;
    voa;
    };
```

Fig. 8B

```
class AAMatrix : public...
    {
    aa[][nStage+1];
    h;
    voa;
    };
```

Fig. 9

```
EvaluateXAllocAgainstWScenario(xAlloc, wScenario)
    {
    AAMatrix aaMatrix;
    copy xAlloc.a1 to aaMatrix.aa[][1];

    for(iFlexStage=2;iFlexStage<=nStage;iFlexStage++)
        {
        DeterministicOptimizer(iFlexStage, aaMatrix,
          wScenario.pWNode[iFlexStage]->pGBS->wwMatrix);
        }
    xAlloc.voa = aaMatrix.voa;
    }
```

Fig. 10

```
DeterministicOptimizer(iFlexStage, aaMatrix, wwMatrix)
   {
   Assuming that the allocations in aaMatrix for
   stages prior to stage iFlexStage are fixed, and
   that the scenario specified in wwMatrix occurs
   with certainty:
       {
       Apply prior-art techniques to optimize
       allocations for stages iFlexStage to
       nStage.
       }

   Transfer allocations for stage iFlexStage to
   aaMatrix.aa[][iFlexStage]. Transfer
   allocations for stages beyond iFlexStage to
   corresponding columns in aaMatrix.aa.

   ValueAllocation(aaMatrix, wwMatrix);
   }
```

Fig. 11

```
ValueAllocation(aaMatrix, wwMatrix)
    {
    Assess wwMatrix together with aaMatrix.
    Determine an assessed value indicative
    of the desirably of having wwMatrix and
    aaMatrix occur together, as opposed to
    any other pair occurring.

    Set wwMatrix.voa = assessed value;
    Set aaMatrix.voa = assessed value;
    }
```

Fig. 12

```
class ZCluster : public...
    {
    wScenarioCntn;
    xAllocBestCntn;
    xAllocOpt;

    XAllocEvaluator(xAlloc, OKadd);
    ConsiderAppendBestXAlloc(xAllocAdd, xAllocDel);

    Improver();
    SimpleParabolaSearch();
    InnerCompressSearch();
    OuterCompressSearch();
    GenxAlloch(h);
    xAllocHurdle;
    wasImproved;
    xAlloc0;
    xAlloc0off;
    xAlloc1;
    xAlloch;
    xAllocBnd;
    };
```

Fig. 13A

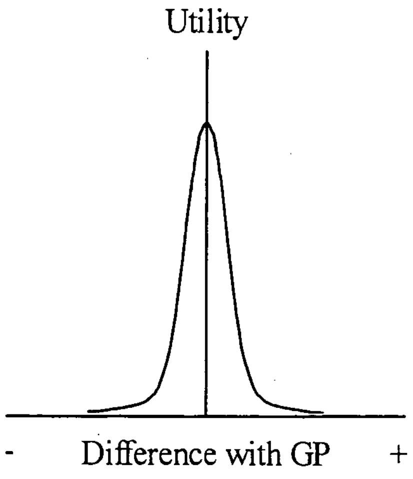
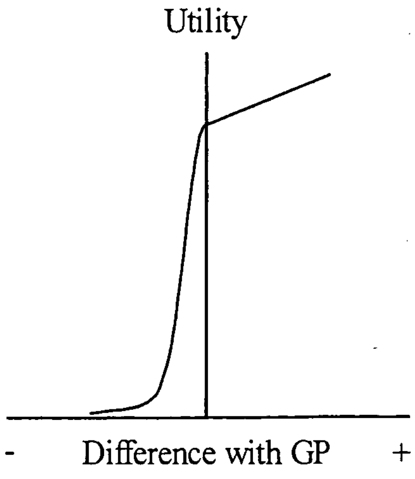


Fig. 13B


Utility
Difference with GP